REMARKS

The claims are claims 1 to 18.

Claims 1, 8, 10, 11 and 18 are amended. Claims 1 and 10 are amended to add a limitation that the operating mode is assigned to fetch packets. Assignment of the operating mode on the basis of fetch packets was not clear from the original language. Claims 8 and 18 are amended to add the limitation that the added no operation machine words are added due to differing instruction latencies between the base and migrant architectures. Claim 11 ahs been amended to correct a grammatical error.

Claims 1, 3, 7, 10, 12, and 16 were rejected under 35 U.S.C. 103(a) as made obvious by the combination of Shiell et al U.S. Patent Number 5,961,632 and O'Connor U.S. Patent Number 5,848,288.

Claims 1 and 10 recite subject matter not made obvious by the combination of Shiell et al and O'Connor. Claims 1 and 10 recite "assigning each fetch packet an operating mode in dependence upon the execution mode at the time the request was made to the memory for the fetch packet." This assigns the operating mode in a different manner than that disclosed in Shiell et al. The FINAL REJECTION fails to point out any portion of Shiell et al disclosing the selection of operating mode based upon the claimed fetch packets. Shiell et al teaches at column 4, line 27 to column 6, line 37 and illustrate in Figures 2a and 2b that mode is dependent upon an instruction leading code. Shiell et at states at column 4, lines 35 to 39:

> "The instruction path leading code may be in different formats, but in any event is preferably set forth in the instruction stream so as to distinguish one or more instructions following the instruction path leading code and to accomplish the functionality set forth below."

PAGE 9/13 * RCVD AT 2/27/2004 10:02:10 AM [Eastern Standard Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:972 917 4417 * DURATION (mm-ss):04-36

Setting the mode by the instruction leading code of Shiell et al differs from claims 1 and 10 in two regards. First, Shiell et al makes clear that his instruction leading code "is preferably set forth in the instruction stream," either as an instruction prefix included within an instruction (column 4, lines 40 to 42) or an escape code that precedes the instruction (column 4, lines 48 to 50). The language of claims 1 and 10 do not require the mode to be set by bits within the instruction stream as required by the teaching of Shiell et al. Second, Shiell et al teaches the mode is operable for "one or more instructions following the instruction path leading code." Thus Shiell et al provides the possibility that plural instructions within a fetch packet such as described in this application can have differing modes. In contrast, in the language quoted above from claims 1 and 10 the operating mode is assigned to the fetch packet. Later recitations within claims 1 and 10 make clear that all instructions within a fetch packet are treated as having the same operating mode. Accordingly, claims 1 and 10 are allowable over Shiell et al.

Claims 2, 4 to 6, 8 to 9, 11, 13 to 15, 17 and 18 were rejected under 35 U.S.C. 103(a) as made obvious by the combination of Shiell et al U.S. Patent Number 5,961,632, O'Connor U.S. Patent Number 5,848,288 and Nishioka et al U.S. Patent Number 6,401,190.

Claims 2 and 11 recite subject matter not made obvious by the combination of Shiell et al, Connor and Nishioka et al. Claim 2 recites "a third input to said multiplexer wherein said third input is a no operation machine word." Similarly, claim 11 recites "choosing between the output of said migrant architecture decode and the output of said base architecture decode input and a no operation machine word." Respective base claims 1 and 10 make clear that the machine words result from decoding of individual instructions in the respective base mode or migrant mode. The combination of references fail to make obvious the third choice of

PAGE 10/13 * RCVD AT 2/27/2004 10:02:10 AM [Eastern Standard Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:972 917 4417 * DURATION (mm-ss):04-36

a no operation machine word recited in claims 2 and 11. Figure 1 of Shiell et al illustrates multiplexer 24 selecting between paths 18, 20 and 22. Shiell et al states "each of paths 18, 20, and 22 represents a path through which a fetched instruction may pass" (Shiell et al at column 3, lines 7 to 9). Figures 3a and 3b of Shiell et al illustrate these instruction paths 18, 20 and 22 in further detail. In Figures 3a and 3b Shiell et al discloses obtaining executable instruction words from: microprogram memory 18g; a CISC instruction path including predecode stages 18a and 18b followed by decode stages 18c and 18d; a RISC instruction path including decode stages 20a and 20b; and a direct path 22a. The Applicants respectfully submit that these inputs are equivalent to the output of the base architecture decode and the output of the migrant architecture decode recited in claims 2 and 11. None of these inputs are the no operation machine word recited in claims 2 and 11. The FINAL REJECTION cites column 7, lines 31 to 59, column 8, line 47 to column 9, line 20, column 14, lines 30 to 37, Figure 5; and Figure 6 of Nishioka et al as allegedly making this third no operation selection from the multiplexer as obvious. However, these cited portions of Nishioka et al fail to disclose any such multiplexer or selection of machine words from alternative sources. The nop fields illustrated in Figures 5 and 6 of Nishioka et al are instructions within an expanded execute packet formed by instruction expansion circuit 2. Nishioka et al states at column 14, lines 30 to 37 (cited in the FINAL REJECTION):

> "As a specific example, FIG. 5 shows the header 0 shown in FIG. 3 and the corresponding expanded instruction formats 1 through 4. It is assumed that these four instructions be all specified with the normal mode. The expanded instruction format is generated by the instruction expansion circuit 2 from the field presence/absence information. In the instruction memory 1, the omitted NOP field is generated and the fields are sorted."

PAGE 11/13 * RCVD AT 2/27/2004 10:02:10 AM [Eastern Standard Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:972 917 4417 * DURATION (mm-ss):04-36

The Applicants respectfully submit these NOP fields are the result of decoding the original instruction input. Thus these NOP fields together with other decoded operations correspond to the output of said migrant architecture decode and the output of said base architecture decode as recited in base claim 1 and claim 11. Accordingly, these NOP fields do not make obvious the additional no operation machine word selection recited in claims 2 and 11. Accordingly, claims 2 and 11 are allowable over the combination of Shiell et al and Nishioka et al.

Claims 8 and 18 recite subject matter not made obvious by the combination of Shiell et al, Connor and Nishioka et al. Claim 8 recites "said migrant architecture control circuit for issuing no-operation instruction to preserve the semantics of the instruction in the migrant architecture due to differences in instruction latencies between the base architecture and the migrant architecture." Similarly, claim 18 recites "issuing no-operation instruction from said migrant architecture control circuit, to preserve the semantics of the instructions in the migrant architecture due to differences in instruction latencies between the base architecture and the migrant architecture." Nishioka et al teaches provision of nop instructions for a different purpose than recited in claims 8 and 18. Nishioka et al states at column 28, lines 40 to 45:

> "Normally, in VLIW, about 80% of the objects is occupied by NOP. Therefore, NOP compression is an essential technology when memory usage efficiency is taken into consideration. Use of the header used by this technology also in the SIMD mode mitigates the overhead, which is the feature of this embodiment."

This portion of Nishioka et al states that nops are inserted for compression and memory usage. This is not the "to preserve the semantics of the instruction in the migrant architecture due to

PAGE 12/13 * RCVD AT 2/27/2004 10:02:10 AM [Eastern Standard Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:972 917 4417 * DURATION (mm-ss):04-36

differences in instruction latencies between the base architecture and the migrant architecture" as recited in claims 8 and 18. Accordingly, claims 8 and 18 are not made obvious by the combination of Shiell et al, Connor and Nishioka et al.

Claims 3 to 7 and 9 are allowable by dependence upon allowable base claim 1. Claims 12 to 17 are allowable by dependence upon allowable base claim 10.

The Applicants respectfully submit that all the present claims are allowable for the reasons set forth above. Therefore early entry of this amendment, reconsideration and advance to issue are respectfully requested.

If the Examiner has any questions or other correspondence regarding this application, Applicants request that the Examiner contact Applicants' attorney at the below listed telephone number and address to facilitate prosecution.

Texas Instruments Incorporated
P.O. Box 655474 M/S 3999
Dallas, Texas 75265
(972) 917-5290
Fax: (972) 917-4418

Respectfully submitted,

Robert D. Marshall, Jr.
Reg. No. 28,527

PAGE 13/13 * RCVD AT 2/27/2004 10:02:10 AM [Eastern Standard Time] * SVR:USPTO-EFXRF-1/2 * DNIS:8729306 * CSID:972 917 4417 * DURATION (mm-ss):04-36